Xacct
Internet accounting

XaCCT 3.0

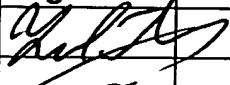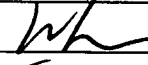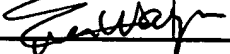## *System Specification Document*

Identification XACCT-SSD

Revision 1.0

July 14, 1997

Signatures

	Name	Title	Signature	Date
Written By:	Tal Givoly	SW Team Leader	*[signature]*	14/7/97
Checked By:	Yuval Tal	SW Team Leader	*[signature]*	14/7/97
Checked By:	Limor Schweitzer	SW Dev. Manager	*[signature]*	15/7/97
Approved By:	Eran Wagner	President	*[signature]*	27/7/97

Changes

Revision	Date	Name	Description

TABLE OF CONTENTS

1. GENERAL

1.1. SCOPE

This document contains the system specifications of the XaCCT 3.0 system. The document identifies the system components (CSCIs), the top-level relationships between these components and the scope of each component.

1.2. APPLICABLE DOCUMENTS

[1] XaCCT 3.0 Vision Statement, XACCT-VS, Revision 1.0, dated June 23, 1997.

[2] XaCCT 3.0 White Paper, Revision 1.4, dated March 18, 1997.

[3] Central Event Manager Software Requirement Specification, CEM-SRS, Revision 1.0.

[4] Gatherer Software Requirement Specification, GATHERER-SRS, Revision 1.0.

[5] Configuration User Interface Software Requirement Specification, CUI-SRS, Revision 1.0.

[6] Reports User Interface Software Requirement Specification, RUI-SRS, Revision 1.0 (TBD).

[7] Gatherer-CEM Interface Description Document, Gatherer-CEM-IDD, Revision 1.0 (TBD).

[8] CUI-CEM Interface Description Document, CUI-CEM-IDD, Revision 1.0.

[9] Unified Network Information Record (UNIR) Interface Description Document, UNIR-IDD, Revision 1.0.

[10] Gatherer-Gatherer Interface Description Document, Gatherer-IDD, Revision 1.0 (TBD).

In any case of discrepancy between any of the above documents and this document, this document will take precedence

1.3. SYSTEM OVERVIEW

The XaCCT 3.0 system is designed to provide network accounting and billing information for usage of various network resources. The resulting accounting and billing information may be used by corporate network management for internal accounting of

The network resource consumption information is extracted from several information sources, such as FireWall-I logs, Web logs, Mail logs, routers, etc.. Furthermore, it is expected that the list of information sources to be monitored for information regarding resource consumption will grow.

XaCCT 3.0 is expected to support a wide array of such information sources, and enable generating accounting and billing reports based on as much information as it can access. This information will be cross-referenced between information sources to remove duplicates, and to augment information collected from the various sources.

In order to support such a wide array of information sources, a generic approach to accessing an information source shall be used. Information sources are accessible from an Gatherer module (or Information Source Module - ISM) that is close to the information source and interacts with it directly. The Gatherer then conveys the information gathered from this information source to a central database, managed by a Central Event Manager (CEM). Either the CEM or the Gatherer may augment the data collected with data from other information sources before placing in database. The information collected can be accessed later on using a report generator tool called the Reports User Interface (RUI) component. The whole system is configured using a Configuration User Interface (CUI).

TBD: Insert here the graph from the white paper and describe it.

1.4. GLOSSARY OF TERMS AND ABBREVIATIONS

AISM   Asynchronous Information Source Module

CDB   Central Database

CEM   Central Event Manager

CSCI   Computer Software Configuration Item. An aggregation of software designated for configuration management and treated as a single entity.

CUI   Configuration User Interface

DB   Data Base

DBMS   Data Base Management System

HTML   Hyper-Text Markup Language

IDD   Interface Description Document

IS   Information Source

ISCM   Information Source Configuration Module

ISM   Information Source Module

ISP   Internet Service Provider

LAN   Local Area Network

LMS   Local Module Storage

MMI   Man-Machine Interface.

OS   Operating System

RUI   Reports User Interface

SISM   Synchronous Information Source Module

SRS   Software Requirements Specification

TBD   To Be Defined

UDP   User Datagram Protocol

UNIR   Unified Network Information Record

2. MAIN COMPONENTS (CSCI)

The system is composed of several CSCIs described in this section. These CSCIs interact using the interfaces described in "4. Internal Interfaces Between CSCIs".

Below is a schematic image of the CSCIs in the system and external entities they interact with:

```
┌──────────────┐    ┌──────────────┐    ┌──────────────┐
│ Information   │    │ Information   │    │ Information   │
│ Source        │    │ Source        │    │ Source        │
└──────┬───────┘    └──────┬───────┘    └──────┬───────┘
       │                   └─────────┬──────────┘
       ▼                             ▼
┌──────────────┐            ┌──────────────┐
│              │            │              │
│   Gatherer   │            │   Gatherer   │
│              │            │              │
└──────┬───────┘            └──────────────┘
       │
       ▼
┌──────────────────┐   ┌──────────────────────┐   ┌──────────────────┐
│ Central Event    │   │ Configuration User   │   │                  │
│ Manager          │◄─►│ Interface            │◄─►│  HTML Browsers   │
│ CEM              │   │ CUI                  │   │                  │
└────────┬─────────┘   └──────────────────────┘   └──────────────────┘
         ▼
   ┌──────────┐        ┌──────────────────────┐   ┌──────────────────┐
   │ Central  │        │ Reports User         │   │                  │
   │ Database │───────►│ Interface            │◄─►│  HTML Browsers   │
   │ CDB      │        │ RUI                  │   │                  │
   └──────────┘        └──────────────────────┘   └──────────────────┘
```
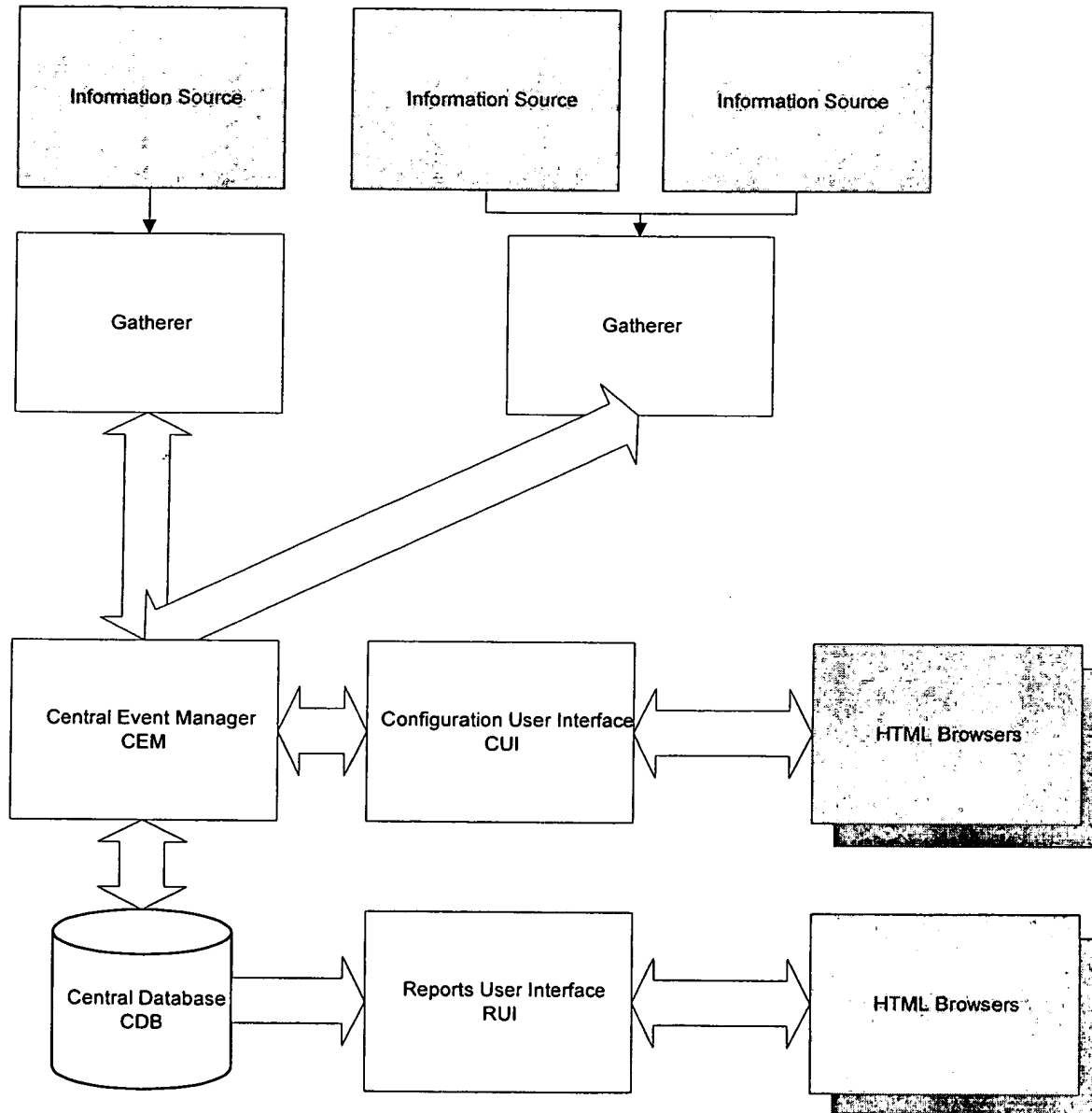
Figure 1: Top-level System Block Diagram

Note: The elements illustrated as white blocks are CSCIs in the system. The gray blocks are other components the CSCIs interact with. These components include the HTML Browser and the Information Sources.

2.1. CEM - CENTRAL EVENT MANAGER

The Central Event Manager CSCI shall be identified as CEM. For detailed information, please refer to [3].

The CEM is responsible for coordinating all system activity. These activities include the configuration and monitoring of the Gatherers and the Information Sources they are connected to and the access to the database to store the network transactions.

The CEM also functions as a license manager. Licenses for Information Sources or Gatherers (TBD) are installed on the CEM, and only the installed number of licenses may be configured to run simultaneously.

The CEM is required to be fault tolerant, i.e., it is expected to be able to recover from any type of system crash. Upon startup of the CEM it will read from the database the system settings, and coordinate the recovery of the system to its defined state.

The CEM is responsible (with possible aid of the Gatherers) to consolidate information and to avoid duplicate entries for the same network transactions. It also is responsible for performing clean-up activities of the database.

The CEM runs on one server running either Windows NT 4.0 (initially on Intel x86 machines only) or Solaris 2.5.

The server should be powerful enough (TBD - elaborate and be more specific) to handle the collection of all the installed Gatherers and Information Sources.

The CEM is connected to one DBMS for storing the information collected from the Gatherers. It is also responsible for maintaining information regarding the installed licenses and other system configuration information. It may use another proprietary database to store this information locally, or store this information in the same DBMS it uses to store the collected information (TBD).

Only one instance of the CEM may be run in a particular system. This is to allow easy central administration of the entire system.

The CEM may be remotely adminstrated by any Web browser. This administration is performed via the CUI.

The CEM also supports time synchronization of the Gatherers in order to assure that all components in the system use a synchronized clock.

The CEM is also responsible to upgrade the Gatherers' version in the field automatically. New versions of the Gatherers' executables are installed on the CEM and these are upgraded automatically by the Gatherers and the CEM. Automatic

upgrading also relates to Information Source related modules that are automatically upgraded by the CEM.

2.2. GATHERER

The Gatherer CSCI shall be identified as Gatherer. For detailed information, please refer to [4].

The Gatherer is responsible for gathering information from one or more Information Sources (ISs) and relaying some consolidation of this information to the CEM for storage. The information is maintained in a generic Unified Network Information Record (UNIR).

The Gatherer is supposed to be installed close to the IS in order to minimize the effect of the communications between the Gatherer and its associated ISs in terms of network bandwidth required and CPU usage.

Multiple Gatherers may report to a single CEM but a Gatherer reports to one CEM only. This CEM is also responsible for monitoring and configuring the Gatherer.

A Gatherer may be accessed by other Gatherers to supply them with access to ISs connected to the Gatherer, upon demand.

The Gatherer has no user interface. Monitoring and configuring a Gatherer is done using any HTML browser and accessing the CUI. The CUI then accesses the CEM which in turn accesses the Gatherer itself.

The configuration of the Gatherer is first done in collaboration between the CUI and the CEM. Once the CUI configures a Gatherer, it is the CEMs responsibility to maintain this configuration, and have it available on demand when a Gatherer is restarted.

The Gatherer is normally a light-weight task that doesn't interfere with normal operation of the computer or network in which it resides, but in some special circumstances, it might run on a dedicated computer or network. The consumption of computer and network resources by the Gatherer itself are also configured by the CEM.

The Gatherer is supposed to be highly portable because it is installed in close proximity to the ISs it collects information from, and the computers existing in these locations may be of different platforms. Initially, Windows 95, Windows NT (both x86 based) and Solaris will be supported, but more platforms will need to be supported in the future.

The Gatherer is also supposed to be generic, in the sense, that it is capable of communicating with a wide variety of ISs.

Since there may be many Gatherers in a particular system installation, and the Gatherers themselves are distributed within this system, it is desirable to be able to upgrade the Gatherers software without need to upgrade the Gatherers individually and manually. Therefore a mechanism for automatic collective upgrade of the Gatherers shall be developed.

The Gatherer can handle many requests by different parties simultaneously, in a manner that one request doesn't interfere with the handling of another requests (possibly using multiple threads).

The CEM may command the Gatherer to collect information and/or relay information in a particular path either to another Gatherer or to the CEM.

The Gatherer is not considered fault-tolerant, and is not responsible in maintaining any information between individual runs. Nonetheless, it may maintain a cache, that may, optionally, be used to speed up the resolution of requests from the CEM.

2.3. CUI - CONFIGURATION USER INTERFACE

The Configuration User Interface CSCI shall be identified as CUI. For detailed information, please refer to [5].

The CUI is an HTTP server that allows multiple users using HTML browsers to configure the XaCCT 3.0 system and to monitor its state. Specifically, the CUI supplies the user interface for both the Gatherers and the CEM.

The CUI provides user interface to register a new Gatherer, to un-register a registered Gatherer, to view the list of registered Gatherers, and to monitor and configure a particular Gatherer. Configuring a Gatherer includes configuring the ISs it collects information from. This configuration is very dependent on the type of the IS being configured, therefore a generic approach to configuring an IS will be applied. Actually, the CUI generates the user interface to configure, monitor and control an IS using parameters it receives from the CEM, or from an IS Configuration Module (ISCM) specific to the IS it is configuring, monitoring or controlling.

The CUI provides user interface to monitoring the CEMs state and to configuring the CEM. This includes setting up the required data to be stored, monitoring the usage of licenses, monitoring the state of the Gatherers associated with the CEM, the CEM and Gatherers' load, operation status indications, etc.

One instance of the CUI exists in the system. The CUI may, optionally, be installed on the same computer on which the CEM is installed. In case of extremely highly loaded networks it might be desirable to install the CEM and CUI on different computers. In this case, the CUI communicates with the CEM over a high-bandwidth TCP/IP connection.

The CUI shall be capable of running on any platform the CEM can run on.

The CUI itself shall produce a local activation log, and otherwise, present no user interface or administration facilities.

2.4. RUI - REPORTS USER INTERFACE

The Reports User Interface CSCI shall be identified as RUI. For detailed information, please refer to [6].

The RUI is an HTTP server that allows multiple users using HTML browsers to design and generate reports based on information collected by the XaCCT 3.0 system.

This reporting tool is used to actually generate the accounting and billing reports required by the user. The rest of the system is responsible for collecting the information on which these reports are based.

This tool is designed to be independent of the rest of the system so that alternative reporting tools may be used to take advantage of the information collected by the XaCCT 3.0 system by simply accessing the information stored in the database.

Reports may be generated either as HTML documents, formatted text, unformatted text or printed documents. Also, graphical representations (e.g. pie charts) will be supported. Some reports may be continuously updated to present new information. These updating reports will only be available as dynamically updating HTML pages.

The report definition process involves selecting **what** information to report, and **how** to render that information as a report. What information to report is defined using some high-level, generic querying facility (higher level than SQL). How the report is to be rendered is defined using another high-level, generic report construction tool.

The report definitions are, themselves, maintained locally to the RUI, or stored in the central database (TBD - update graph arrow if stored centrally).

The RUI shall interrogate the database for its structure, and the setup of ISs configuration as set-up by the CUI, and use this information to limit the user's selection of fields and summaries accordingly.

The RUI shall be capable of running Windows 95, Windows NT (only x86 based) and Solaris (TBD).

The RUI itself shall produce a local activation log, and otherwise, present no user interface or administration facilities.

2.5. CDB - CENTRAL DATABASE

The Central Database CSCI shall be identified as CDB.

The CDB is the central repository of information collected by the XaCCT 3.0 system. The CEM is responsible for generating and maintaining this information, while the RUI is responsible for utilizing this information to generate accounting and billing reports. More than one reporting tool (such as RUI) may be connected simultaneously to generate reports based on the collected information.

The CDB may be run on several commercial SQL DBMSs, including MS-SQL Server, Oracle SQL Server, and Informix SQL Server. The DBMS is assumed to be fault-tolerant, and to be capable of recovery after system crashes. This means that all committed transactions performed on the database will survive a system crash, while all uncommitted or rolled-back transactions do not.

The CDB structure will be configured by the CEM upon installation.

One CDB shall exist for each XaCCT 3.0 system.

3. EXTERNAL COMPONENTS

The following components interact directly with the XaCCT 3.0 system.

3.1. INFORMATION SOURCES

Information Sources shall be identified as ISs.

An IS is an instance of any source of network information. This includes routers, Web servers, FireWalls, Mail servers, Name resolution servers (e.g. DNS), Authentication servers, etc..

It is assumed that each type of information source may provide information in a different method or protocol. Some may generate logs, others may be accessible via SNMP, yet others may have proprietary APIs or any other proprietary protocol.

Multiple ISs may be accessed by a Gatherer and an IS may be accessed by multiple Gatherers (so long as they do not collide - this is the responsibility of the ??? TBD).

3.2. EXTERNAL REPORT GENERATORS

External Report Generators shall be identified as XRGs.

XRGs may generate reports based on data collected by the XaCCT 3.0 system by accessing the CDB on which the data has been collected. These report generators may be produced by third party software developers and VARs.

XRGs will rely upon an exposed interface to the CDB published by XaCCT to expose the internal DB structure or queries.

No XRGs are currently available.

3.3. HTML BROWSER

The HTML Browser shall be identified as HTMLB or HTML Browser or Web Browser.

An HTML browser is used as the sole user interface front-end of the XaCCT 3.0 system. No platform-dependent user interfaces shall be used.

The HTML browser is used for the CUI front-end and the RUI front-end.

The HTML browser is assumed to support HTML 3.2 including frames and is assumed to have a Java Virtual Machine capable of running Java 1.0 applets.

Currently Microsoft Internet Explorer, Netscape Navigator, and HotJava, are all known to support the above requirements of an HTML browser.

4. INTERNAL INTERFACES BETWEEN CSCIS

This section describes all the internal interfaces identified in the system overview.

4.1. GATHERER- CEM INTERFACE

Enclosed is a brief description of this interface, for a complete description refer to [7].

The Gatherer receives control and command from the CEM, and sends back its status and network information it collected from the ISs it is connected to using this interface.

The interface relies on a TCP/IP connection between the Gatherer and the CEM. The underlying protocol is UNIR. UNIR is completely defined in [9].

This interface is a Mastered interface. The CEM is the Master and controls the operation of the Gatherer. For instance, the CEM may command the Gatherer to Start or Stop collecting information at any time.

This interface is also used to upgrade the Gatherer software when required. Therefore, version checking of Gatherer, Information Source modules, and interface versions are performed at fixed stages in order to verify compatibility and to perform corrective actions (upgrading the software/data version) in case of discrepancy.

4.2. GATHERER- GATHERER INTERFACE

Enclosed is a brief description of this interface, for a complete description refer to [10].

The interface relies on a TCP/IP connection between the Gatherer two communicating Gatherers. The underlying protocol is UNIR. UNIR is completely defined in [9].

This purpose of this interface between any two Gatherers is:
1. To relay messages from any other entity in the system to a third Gatherer. This allows the communication between elements in system to traverse obstacles such as proxies and firewalls.
2. To pass a partially enhanced UNIR for further enhancements at another Gatherer. The sending Gatherer has completed enhancing the UNIR using its local ISs, further enhancements are required at other Gatherers, and the Gatherer sends the complete UNIR (identifying an entire network transaction) to another Gatherer to continue the enhancement process, before finally sending the message to the CEM. More than two Gatherers may be involved in this process, but the message including the partially enhanced UNIR is enhanced sequentially by different Gatherers.
3. To pass requests and replies for information of a local IS to a remote Gatherer. This allows one Gatherer to perform enhancements using information accessed by an IS installed on another Gatherer without passing the whole partially enhanced UNIR to the other Gatherer.

The communication between two Gatherers is completely controlled by the CEM. It is the CEM that instructs a Gatherer to send/receive messages to/from other Gatherers, and what to do with each message (forward, enhance, handle, etc.).

4.3. CUI - CEM INTERFACE

Enclosed is a brief description of this interface, for a complete description refer to [8].

The interface relies on a TCP/IP connection between the CUI and the CEM. The underlying protocol is UNIR. UNIR is completely defined in [9].

The purpose of this interface is to allow the CUI to configure and monitor the XaCCT 3.0 system (mainly CEM, Gatherers, and associated ISs).

The CUI receives through this interface information regarding the system, including its capabilities and its state. And sends configuration requests to the CEM.

The CEM acts as a server whereas the CUI is a client of the configuration and state services provided by the CEM.

4.4. CDB INTERFACES

Both the interface between the CEM and the CDB and the interface between the RUI and the CDB rely on the Open Database Connectivity (ODBC) standard. This standard provides native methods to access databases of different types from different platforms in a standard approach that relies on the fact that a native driver be supplied for the client platform. ODBC provides facilities to access and manipulate data, and to construct and interrogate the structure of the database exposed by the CDB.

Using ODBC is not the only way to access a DBMS using JDBC, but it is most widely supported, therefore, using ODBC has been selected as a preferred method.

5. EXTERNAL INTERFACES

This section identifies all the external interfaces identified in the system overview.

5.1. HTMLB - CUI INTERFACE

This interface is designed to provide remote, platform independent, control and monitoring facilities for the XaCCT 3.0 system.

It is based on Internet standards of HTTP/HTML and Java 1.0 in order to be platform independent.

5.2. HTMLB - RUI INTERFACE

See "5.1. HTMLB - CUI Interface".

5.3. IS-GATHERER INTERFACE

This interface is designed to extract network information from an IS and to collect it via distributed Gatherers. The interface is highly dependent on the particular type of IS, and is proprietary or standard depending on the IS. For each type of IS, the IS-dependent module shall fully describe this interface. Below is a list of classes of interface types that shall be supported:
- TCP client/server.
- UDP client/server.
- SNMP.
- DNS.
- Local file.
- ODBC.
- Command execution.

6. SYSTEM TOP-LEVEL USE CASES

This section contains a collection of top-level/system-level use cases.

6.1. USE CASE: STARTUP & SHUTDOWN (BOOTSTRAPPING)

6.1.1. Description

Overview
Since the system is distributed, the Startup of the system is unique and important to understand. Most generally, any order of loading of components in the system should be supported.

All components are most likely going to be configured to load upon system startup (by being a daemon in Unix systems, a service in Windows NT, and a automatically loaded process in Windows 95). Nonetheless, each such process can be started and stopped independently of others and the computers themselves are never assumed to be synchronized with each other.

Also,..any component may fail and become unavailable at any time.

To support such a distributed installation, all communication links are periodically polled to assess the state of the other party. Also, every component has startup, shutdown and runtime procedures to address the possible disconnected states.
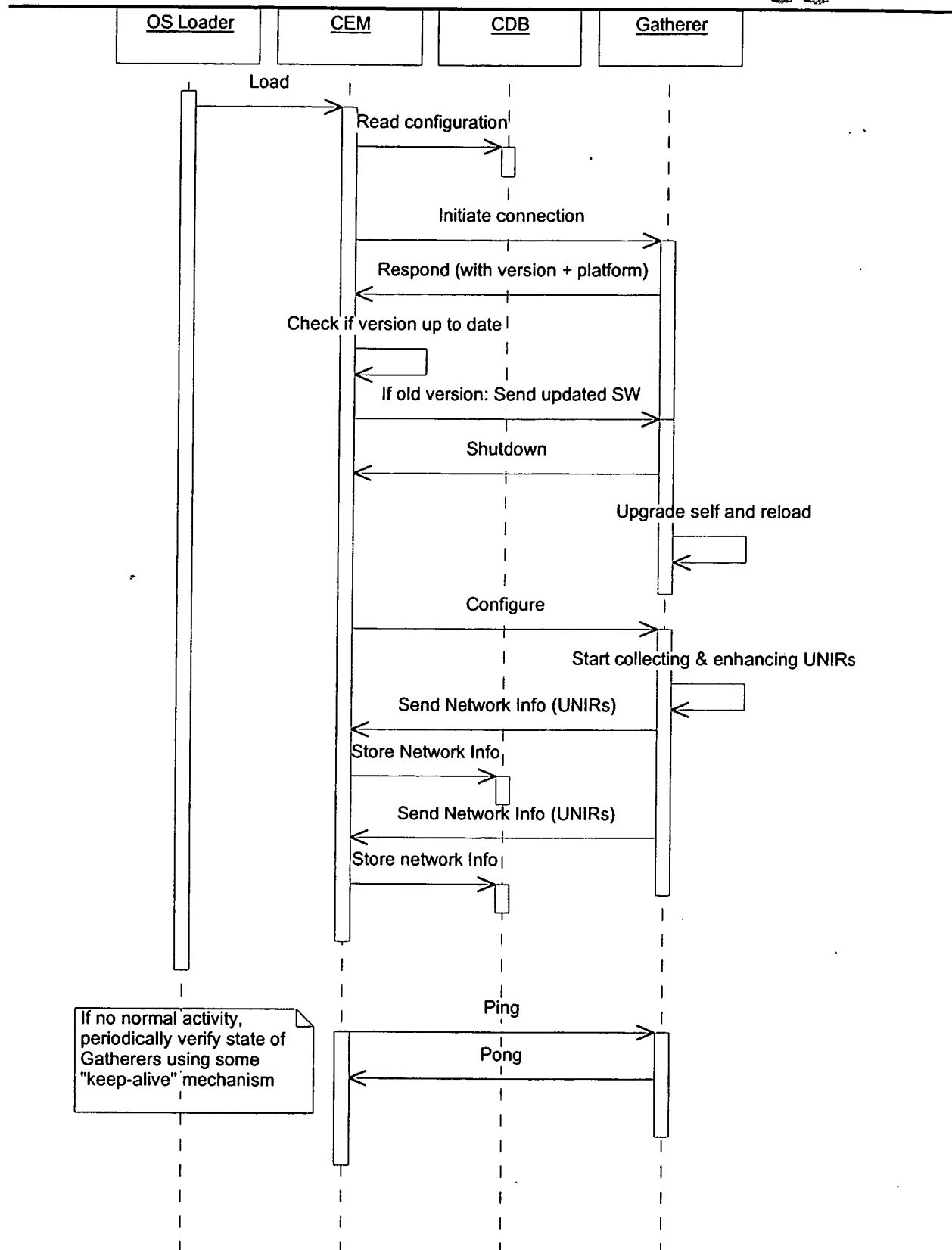
Actor
OS or administrator. The OS will usually load the components installed locally on the computer. An administrator may occasionally start or stop components.
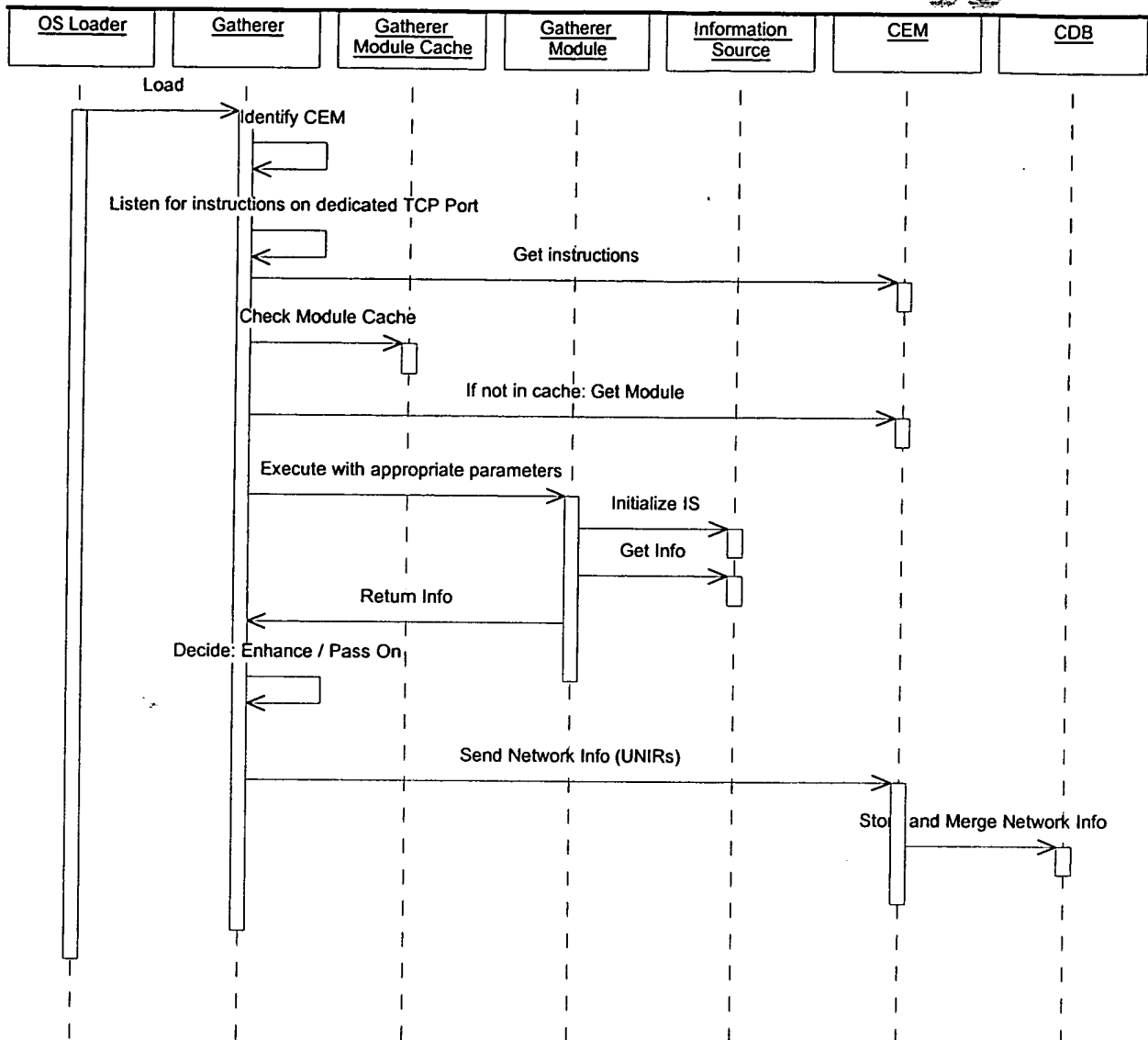
Frequency
High - computers may be powered-up and down or fail regularly.
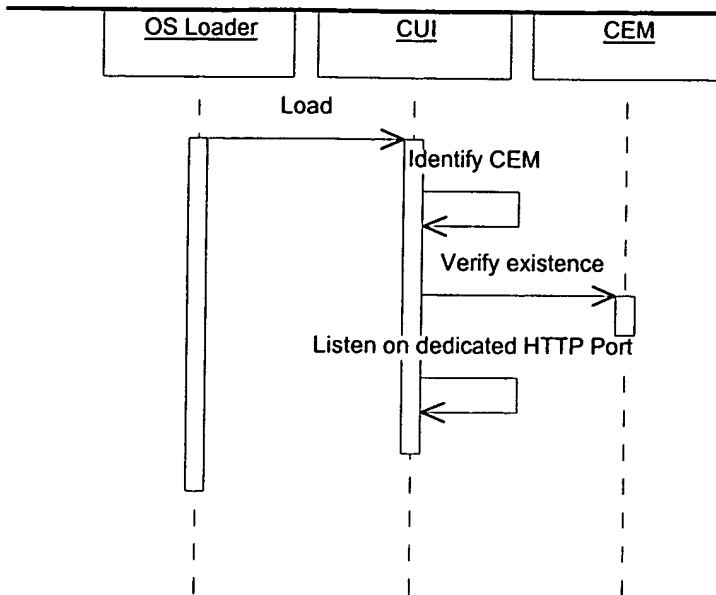
6.1.2. Interaction Diagram

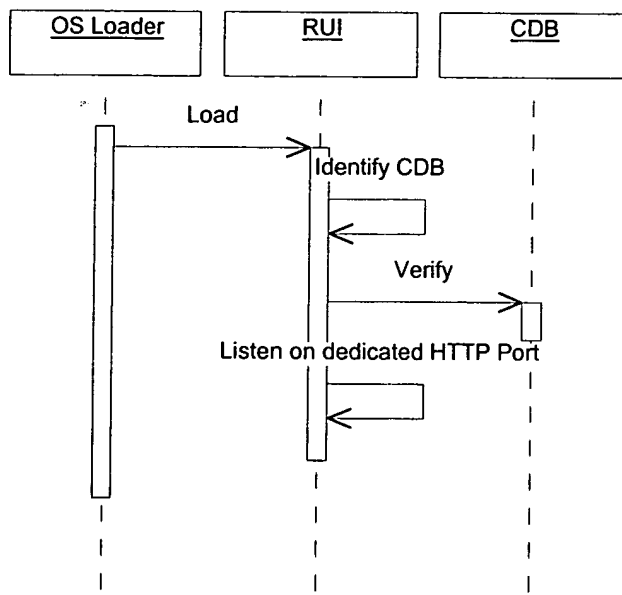Loading of several components are described: CEM, Gatherer, CUI, RUI.

Figure 2: CEM Startup Interaction Diagram

Figure 3: Gatherer Startup Interaction Diagram

The sequence diagram shows the following participants across the top: OS Loader, Gatherer, Gatherer Module Cache, Gatherer Module, Information Source, CEM, CDB.

Message flow:
- OS Loader → Gatherer: Load
- Gatherer: Identify CEM (self-call)
- Gatherer: Listen for instructions on dedicated TCP Port (self-call)
- Gatherer → CEM: Get instructions
- Gatherer → Gatherer Module Cache: Check Module Cache
- Gatherer → CEM: If not in cache: Get Module
- Gatherer → Gatherer Module: Execute with appropriate parameters
- Gatherer Module → Information Source: Initialize IS
- Gatherer Module → Information Source: Get Info
- Gatherer Module → Gatherer: Return Info
- Gatherer: Decide: Enhance / Pass On (self-call)
- Gatherer → CEM: Send Network Info (UNIRs)
- CEM → CDB: Store and Merge Network Info

Figure 4: CUI Startup Interaction Diagram

Figure 5: RUI Startup Interaction Diagram

6.2. USE CASE: CONFIGURATION OF AN INFORMATION SOURCE

6.2.1. Description

Overview
After installing the system, a new Gatherer, or modifying network architecture, reconfiguration of one or more ISs is required.

Actor
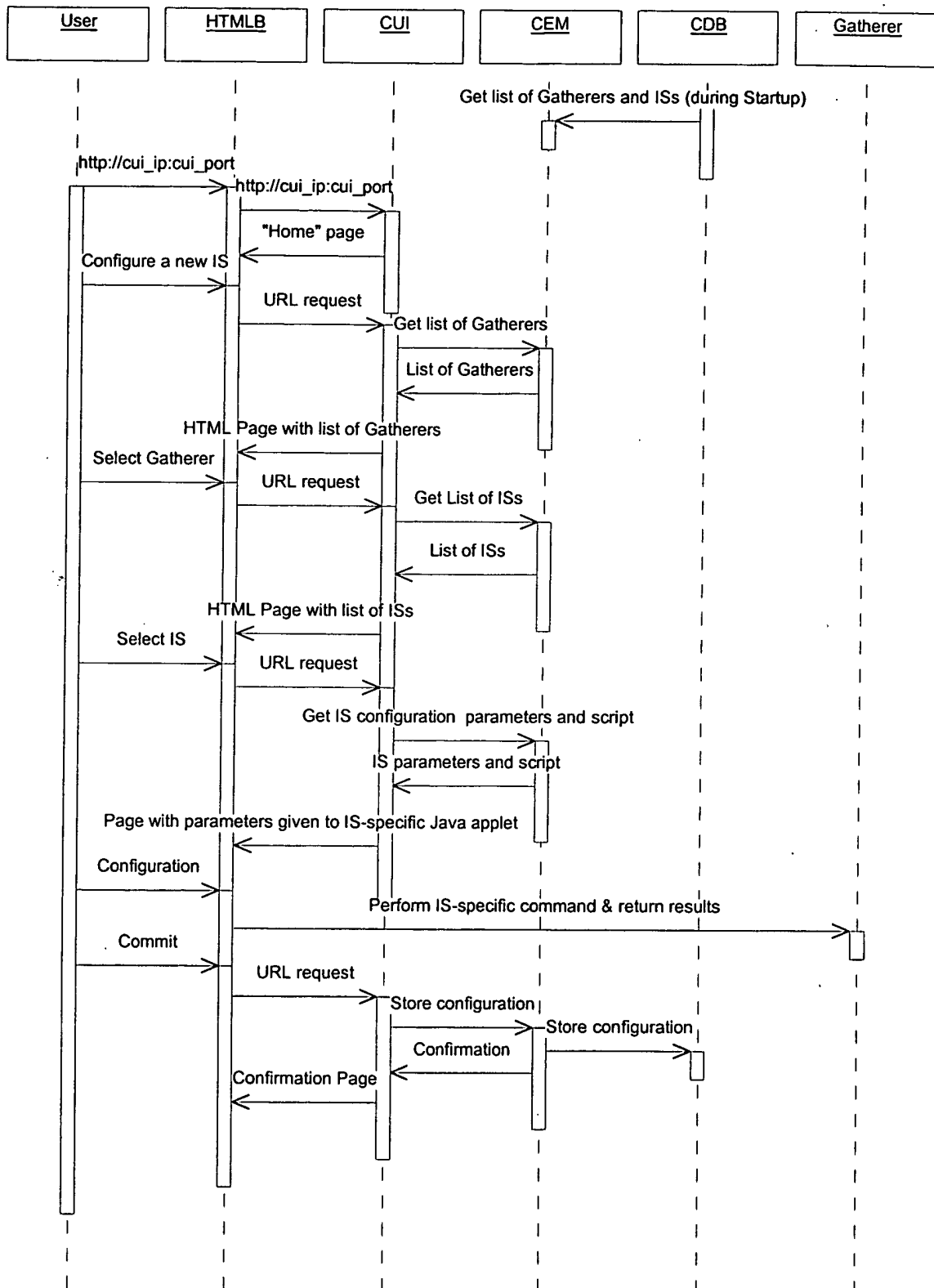
Administrator.

<u>Frequency</u>

Low - performed only upon installation and network architecture changes.

6.2.2. Interaction Diagram

Figure 6: Configuration of IS Interaction Diagram

6.3.1. Description

Overview

Network information is collected by Gatherers from ISs. This information is augmented by other network information collected, possibly, from other ISs, and is sent to the CEM to be placed in the CDB for later retrieval.

The act of collecting network information is initiated by the CEM upon its startup, by commanding the Gatherers to begin to collect information. There is no use in collecting network information while the CEM is not operational. If and when the CEM is shutdown, it commands its Gatherers to stop sending network information.

An extension of this use-case, is the enhancement process, which may involve more than one Gatherer. Enhancing a UNIR (describing a network transaction) may be done in one of the following ways:
1. The Gatherers to which the ISs required to enhance the UNIR are sent the UNIR in its entirety and perform the enhancement and continue to process the UNIR (without returning control to the Gatherer that sent the UNIR.
2. The Gatherer to which the ISs required to enhance the UNIR are sent a request for a particular piece of information, which is later, upon its receipt by the requesting Gatherer, added to the UNIR, and the UNIR's processing continues at the requesting Gatherer.

In both cases, the completely enhanced UNIR is finally sent to the CEM for storage.

Actor
CEM.

Frequency
High - usually performed continuously.

6.3.2. Interaction Diagram

This interaction is described as the continuation of Gatherer Startup in "6.1.2. Interaction Diagram". The extension mentioned above is not described graphically.

6.4. USE CASE: MONITORING SYSTEM STATE

6.4.1. Description

Overview

The fact that most components in the system have no user interface requires some mechanism for monitoring the system state.

The CUI provides these facilities by exposing to the user facilities for remote monitoring of Gatherers and the CEM.

The system state may include the simple indication of liveliness up to the exact parameters of CPU load and information collected by a particular Gatherer. For instance, a packet-capturing ISM may maintain the number of open sessions, network load, average throughput, etc.

The Gatherer and the CEM shall have a fixed set of properties that may be monitored. Besides this fixed set, they may have a variable set of properties that may be viewed, dependent on version, IS, etc. All this information will be exposed to user upon demand via an HTML browser connected to the CUI.

Actor
Administrator.

Frequency
Low - performed only in case of problems.
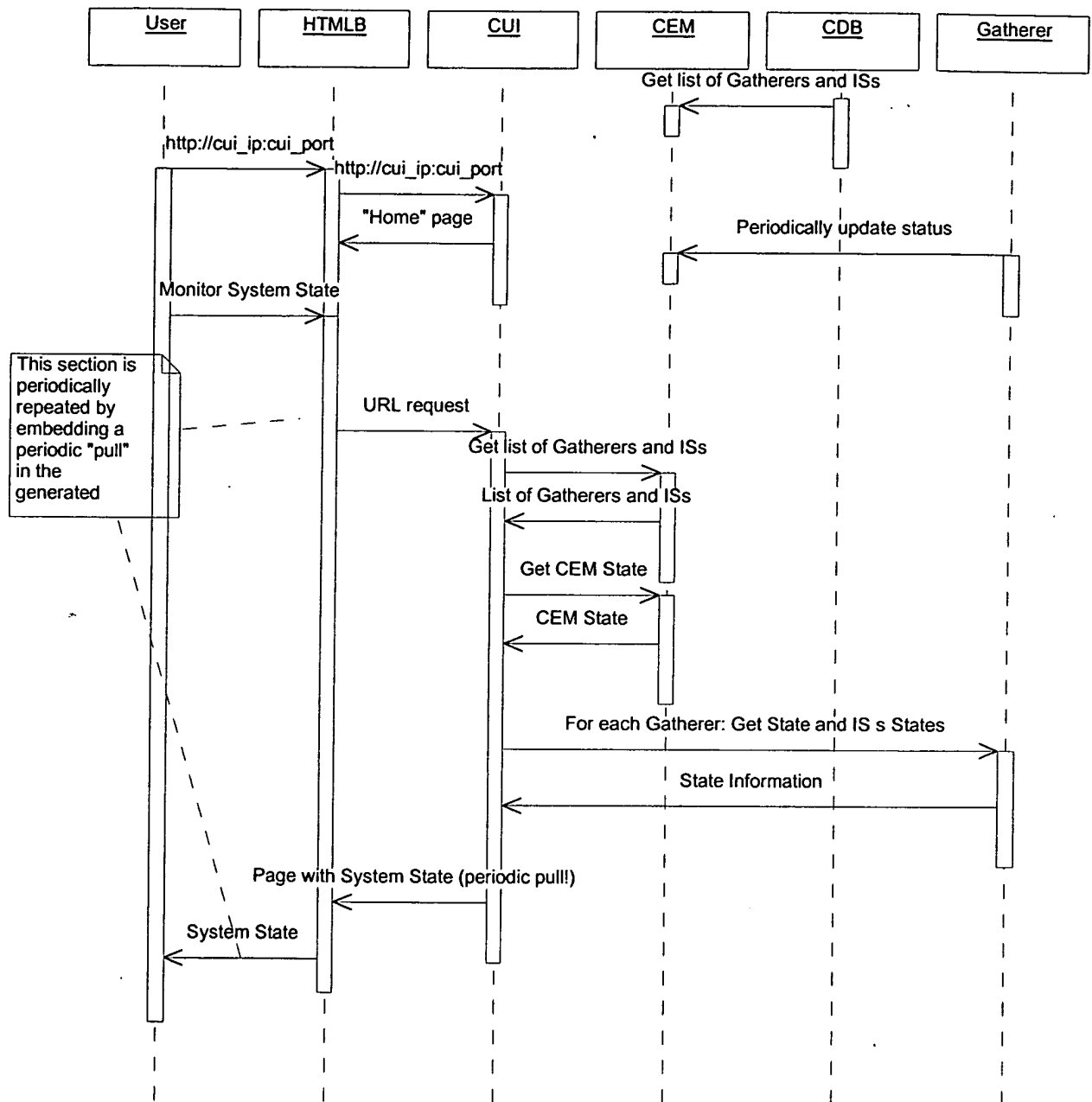
6.4.2. Interaction Diagram

Figure 7: Monitoring System State Interaction Diagram

6.5. USE CASE: GENERATING REPORTS

6.5.1. Description

Overview

The entire system is developed in order to produce reports suitable for accounting and billing of network resource usage. Defining and customizing the

reports for a particular usage is important, but most important is the ability to actually produce the reports in a usable form.

A usable form ranges from a pie-chart or relative consumption to detailed individual bills at the end of the month.

A general process for generating such reports shall be employed.

Also, the automatic generation of such reports is important, since some of these reports should actually be produced periodically.

<u>Actor</u>
User.

<u>Frequency</u>
Medium - this is usually on macroscopic periodic basis (daily, monthly, etc.).
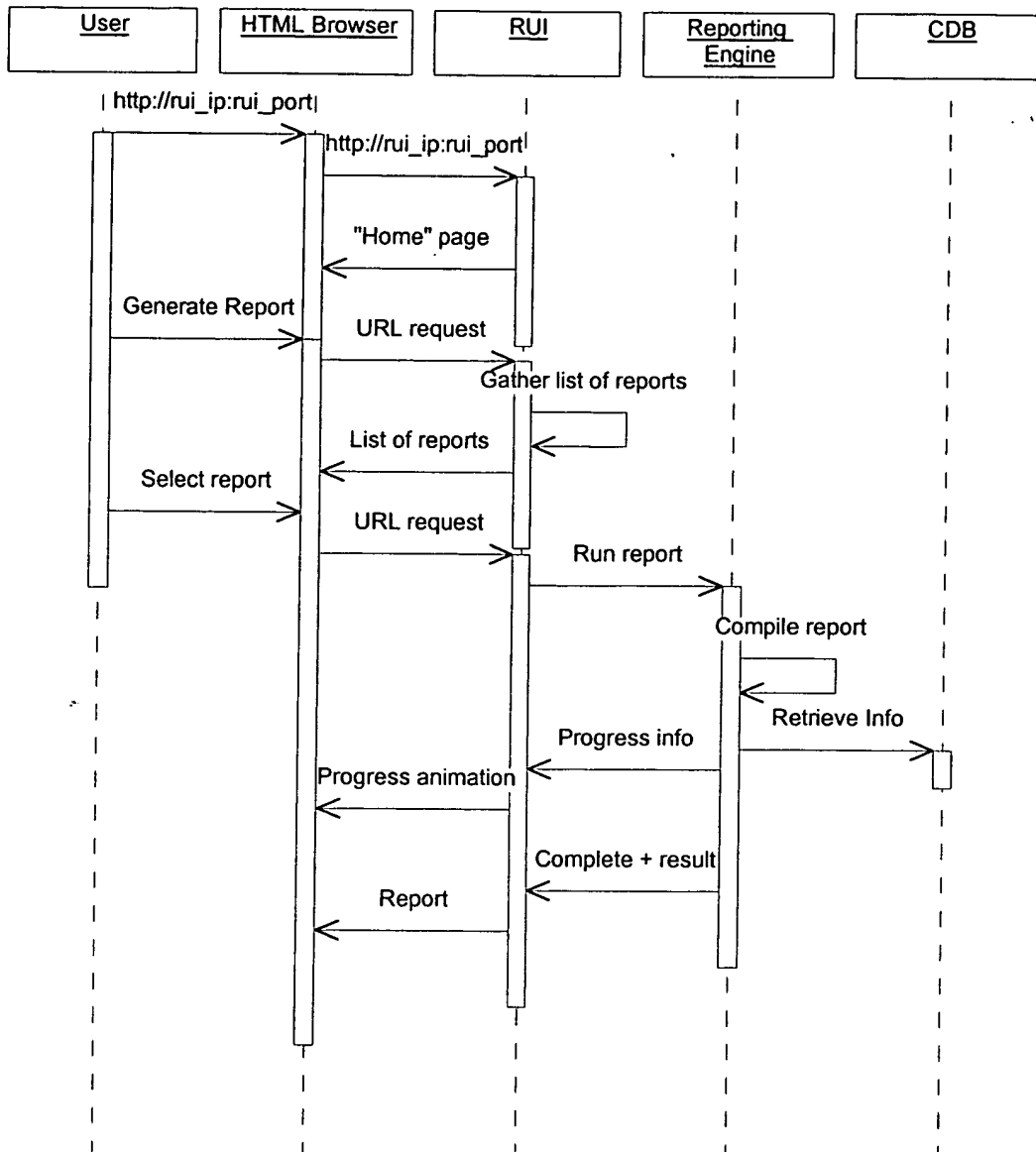
6.5.2. Interaction Diagram

Figure 8: Generating a Report Interaction Diagram